	NOME	OPERANDO 1	OPERANDO 2	Cod. Esad.	DESCRIZIONE
	ALFA	Registro indirizzatore di microprogramma: LØØ→liv. 4 LØ1→liv. 3 A13→liv. 2 A12→liv. 1			Non ha codice esadecimale in quanto è caricata su R0 da hardware. Indirizza la memoria per la lettura di una microistruzione e incrementa di 1 il registro indirizzatore di microprogramma.
PSEUDO MICROISTRUZIONI	RESE	Registro indirizzatore di microprogramma di livello 3: LØ1			Non ha codice esadecimale in quanto è caricata su R0 da hardware. Simula le condizioni di fine fase ∝ e carica in LØ1 la prima microistruzione di inizializzazione (indirizzo esadecimale 8ØØØ).
	NOP			C900	Non opera: in fase ∝ incrementa di 1 il registro indirizzatore del microprogramma in esecuzione: LØØ, LØ1, A13, A12.

	NOME	OPERANDO 1	OPERANDO 2	Cod. Esad.	DESCRIZIONE
MICROISTRUZIONI CHE OPERANO SUI DEVIATORI	SDIB	B(x)		B3xF	Scambia il contenuto dei deviatori con Bx
	SEDI	C(x)(x)		C9xx	Setta i deviatori mascherati da Cxx
	REDI	C(x)(x)		C8xx	Resetta i deviatori mascherati da Cxx
MICROISTRUZIONI TRA GOVERNI DI UNITA' PERIFERICHE E REGISTRI (GOP→Registri)	EDA	A(x)		B8x8	Deposita il canale dati EPD0 ÷ 7 in Ax
	EDB	B(x)		A9x8	Deposita il canale dati EPD0 ÷ 7 in Bx
	ENUA	A(x)		B9x0	Deposita il canale nome EPN0 ÷ 7 in Ax
	ETIB	B(x)		B2xF	Deposita il canale tipo EPT0 ÷ 7 in Bx
	ENTL	L(x)		AAx0	Deposita il canale nome EPN0 ÷ 7 in Ax e il canale tipo EPT0 ÷ 7 in Bx
MICROISTRUZIONI TRA REGISTRI E GOVERNI DI UNITA' PERIFERICHE (Registri → GOP)	DAE	L(x)		FCx0	L(x) viene emesso sul canale dati ECD0 ÷ F accompagnato da ECOT
	CAE	L(x)		FCx2	L(x) viene emesso 2 volte sul canale dati ECD0÷F; la 2ª volta è accompagnata da ECOT ed ECOC
	DEA	L(x)		FBx8	Emette Bx su ECD8 ÷ F accompagnato da ECOT e deposita EPD0 ÷ 7 in Ax
MICROISTRUZIONI DI CONSOLLE (SUCE 2)	TCCA	A(x)		CAx0	Deposita gli 8 bit di consolle CON0 ÷ 7 in Ax
	TDMA	A(x)		DAx1	Trasferisce i dati meno significativi di consolle da CON0 ÷ 7 in Ax
	TDPA	A(x)		EAx2	Trasferisce i dati più significativi di consolle da CON0 ÷ 7 in Ax
	TABC	A(x)	B(y)	FAxy	Trasferisce Ax e By in consolle accompagnato da C010
MICROISTRUZIONI DI COMANDO	COMØ			BD00	Emette CE10N: disattiva il livello in corso, pone a 1 ECORN, non lavora a livello 4
	COM1			BD10	Emette CE11N: attiva il livello 3
	COM3			BD30	Emette CE13N: pone a 0 ECORN resettando i governi periferici
	COM4			BD40	Emette CE14N: fa nascere ECOT
	COM5			BD50	Emette CE15N
	COM6			BD60	Emette CE16N
	COM7			BD70	Emette CE17N: setta EC1F
	COM8			BD80	Emette CE18N: resetta EC1F
	COM9			BD90	Emette CE19N: setta EC2F
	COM1Ø			BDA0	Emette CE20N: resetta EC2F
	COM11			BDB0	Emette CE21N: oscura i display del SUCE2
	COM12			BDC0	Emette CE22N: resetta il bit di start nel SUCE2
	COM13			BDD0	Emette CE23N: nel SUCE 2 segnala lo stop di ALT INDIRIZZATO
	COM14			BDE0	Emette CE24N: predispone un indirizzo di ALT da SUCE2
	COM15			BDF0	Emette CE25N: predispone il SUCE2 a visualizzare sul display "DATA" il dato emesso da UC con una TABC.

	NOME	OPERANDO 1	OPERANDO 2	Cod. Esad.	DESCRIZIONE
Microistruzioni di shift	SLSB	B(x)		D4x1	Shift ad anello, a sinistra, di Bx; il bit più significativo va su D100 e il vecchio contenuto di D100 va sul bit meno significativo di Bx
	SHDA	A(x)		C3x0	Shift a destra di Ax; il bit meno significativo va su D100 e il bit più significativo è posto a 0
	SHDB	B(x)		D3x0	Shift a destra di Bx; il bit meno significativo va su D100 e il bit più significativo è posto a 0
	SLDA	A(x)		C3x1	Shift ad anello, a destra, di Ax; il bit meno significativo va su D100 e il vecchio contenuto di D100 va sul bit più significativo di Ax
	SLDB	B(x)		D3x1	Shift ad anello, a destra, di Bx; il bit meno significativo va su D100 e il vecchio contenuto di D100 va sul bit più significativo di Bx
Microistruzioni di incremento e decremento	ICA	A(x)		85xF	Incrementa di 1 il contenuto di Ax
	ICB	B(x)		95xF	Incrementa di 1 il contenuto di Bx
	ICL	L(x)		A5xF	Incrementa di 1 il contenuto di Lx
	DCA	A(x)		AExF	Decrementa di 1 il contenuto di Ax e posiziona D101: a 1 se il risultato = 0 a 0 negli altri casi
	DCB	B(x)		BExF	Decrementa di 1 il contenuto di Bx e posiziona D101: a 1 se il risultato = 0 a 0 negli altri casi
	DCL	L(x)		E5xF	Decrementa di 1 il contenuto di Lx e posiziona D101: a 1 se il risultato = 0 a 0 negli altri casi
	EDC	L(x)		ADxF	Decrementa di 1 il contenuto dei 12 bit meno significativi di Lx; quando tutti sono a 0 viene emesso ECOF
Microistruzioni di scambio di semi byte	ROTA	A(x)		8BxF	Ruota il contenuto di Ax
	ROTB	B(x)		9BxF	Ruota il contenuto di Bx
Microistruzioni di azzeramento	AZAP	A(x)		BBxF	Azzera i 4 bit più significativi di Ax
	AZBP	B(x)		DBxF	Azzera i 4 bit più significativi di Bx
	AZAM	A(x)		ABxF	Azzera i 4 bit meno significativi di Ax
	AZBM	B(x)		DBxF	Azzera i 4 bit meno significativi di Bx
Microistruzioni di verifica	VRA	A(x)		8ExF	Pone a 1 D101 se il contenuto di Ax è 0
	VRB	B(x)		9ExF	Pone a 1 D101 se il contenuto di Bx è 0
	VRL	L(x)		F5xF	Pone a 1 D101 se il contenuto di Lx è 0
MICROISTRUZIONI CHE OPERANO SUI DEVIATORI	TDIA	A(x)		C5xF	Trasferisce il contenuto dei deviatori in Ax
	TDIB	B(x)		D5xF	Trasferisce il contenuto dei deviatori in Bx
	TADI	A(x)		83xF	Trasferisce Ax nei deviatori
	TBDI	B(x)		93xF	Trasferisce Bx nei deviatori
	SDIA	A(x)		A3xF	Scambia il contenuto dei deviatori con Ax

	NOME	OPERANDO 1	OPERANDO 2	Cod. Esad.	DESCRIZIONE
Microistruzione di salto incondizionato	SAI	nome simbolico		0xxx 13 bit	Salto incondizionato a xxx
Microistruzioni tra memoria e governi di unità periferiche MEMORIA ⟷ GOP	MEI	$M_{04(x)\leq 11}$ $A_{12\leq(x)\leq 15}$		90x0	Invia il byte di memoria indirizzata da Mx (o Ax) al canale dati ECD0 ÷ 7
	MEIP	"		94x0	Invia il byte di memoria indirizzata da Mx (o Ax) al canale dati ECD0 ÷ 7; incrementa di 1 il contenuto di x
	MEIM	"		9Dx0	Invia il byte di memoria indirizzata da Mx (o Ax) al canale dati ECD0 ÷ 7; decrementa di 1 il contenuto di x
	EMI	"		8Dx8	Scrive EPD0 ÷ 7 in memoria indirizzata da Mx (o Ax)
	EMIP	"		A2x8	Scrive EPD0 ÷ 7 in memoria indirizzata da Mx (o Ax); incrementa di 1 il contenuto di x
	EMIM	"		A1x8	Scrive EPD0 ÷ 7 in memoria indirizzata da Mx (o Ax); decrementa di 1 il contenuto di x
	ESE	"		B1x4	Emette la selezione del GOP (ECD0 ÷ 7) da zona di memoria indirizzata da Mx (o Ax); emette ECOS ed ECOT
	ECO	"		B4x2	Emette il comando verso GOP (ECD0 ÷ 7) da zona di memoria indirizzata da Mx (o Ax); emette ECOT ed ECOC
	SEI	"		F1x0	Emette ECD0 ÷ F da zona di memoria indirizzata da Mx (o Ax); emette ECOT
	SEIP	"		F7x0	Emette ECD0 ÷ F da zona di memoria indirizzata da Mx (o Ax); incrementa di 1 il contenuto di x ed emette ECOT
	SEIM	"		FDx0	Emette ECD0 ÷ F da zona di memoria indirizzata da Mx (o Ax); decrementa di 1 il contenuto di x ed emette ECOT
	ESI	"		E0x8	Scrive EPD0 ÷ 7 (o EPT0 ÷ 7) in memoria indirizzata da Mx (o Ax); emette ECOT
	ESIP	"		EBx8	Scrive EPD0 ÷ 7 (o EPT0 ÷ 7) in memoria indirizzata da Mx (o Ax); incrementa di 1 il contenuto di x ed emette ECOT
	ESIM	"		ECx8	Scrive EPD0 ÷ 7 (o EPT0 ÷ 7) in memoria indirizzata da Mx (o Ax); decrementa di 1 il contenuto di x ed emette ECOT
MICROISTRUZIONI CHE OPERANO SU UN REGISTRO					
Microistruzioni di shift	SHSA	A(x)		C4x0	Shift a sinistra di Ax; il bit meno significativo è posto a 0 e il bit più significativo va su D100
	SHSB	B(x)		D4x0	Shift a sinistra di Bx; il bit meno significativo è posto a 0 e il bit più significativo va su D100
	SLSA	A(x)		C4x1	Shift ad anello, a sinistra di Ax; il bit più significativo va su D100 e il vecchio contenuto di D100 va sul bit meno significativo di Ax

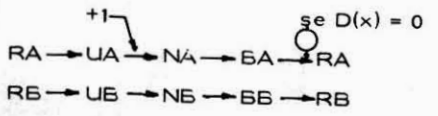
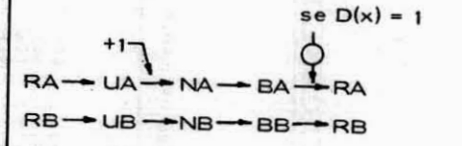
MICROISTRUZIONI CON MEMORIA	NOME	OPERANDO 1	OPERANDO 2	Cod. Esad.	DESCRIZIONE
Microistruzioni con indirizzamento indiretto di memoria	MAI	$\begin{cases} M_{0 \le (x) \le 11} \\ A_{12 \le (x) \le 15} \end{cases}$	A(y)	91xy	Memoria indirizzata da Mx (o Ax) in Ay
	MBI	"	B(y)	99xy	Memoria indirizzata da Mx (o Ax) in By
	MAIP	"	A(y)	98xy	Memoria indirizzata da Mx (o Ax) in Ay con incremento di 1 del contenuto di x
	MBIP	"	B(y)	9Cxy	Memoria indirizzata da Mx (o Ax) in By con incremento di 1 del contenuto di x
	MAIM	"	A(y)	92xy	Memoria indirizzata da Mx (o Ax) in Ay con decremento di 1 del contenuto di x
	MBIM	"	B(y)	9Axy	Memoria indirizzata da Mx (o Ax) in By con decremento di 1 del contenuto di x
	AMI	"	A(y)	A8xy	Ay in zona di memoria indirizzata da Mx (o Ax)
	BMI	"	B(y)	89xy	By in zona di memoria indirizzata da Mx (o Ax)
	AMIP	"	A(y)	88xy	A y in zona di memoria indirizzata da Mx (o Ax) con incremento di 1 del contenuto di x
	BMIP	"	B(y)	8Cxy	By in zona di memoria indirizzata da Mx (o Ax) con incremento di 1 del contenuto di x
	AMIM	"	A(y)	82xy	Ay in zona di memoria indirizzata da Mx (o Ax) con decremento di 1 del contenuto di x
	BMIM	"	B(y)	8Axy	By in zona di memoria indirizzata da Mx (o Ax) con decremento di 1 del contenuto di x
	MLI	"	L(y)	D1xy	Ly è caricato dal contenuto di memoria indirizzata da Mx (o Ax)
	MLIP	"	L(y)	DExy	Ly è caricato dal contenuto di memoria indirizzata da Mx (o Ax); incremento di 1 del contenuto di x
	MLIM	"	L(y)	DDxy	Ly è caricato dal contenuto di memoria indirizzata da Mx (o Ax); decremento di 1 del contenuto di x
	LMI	"	L(y)	E1xy	Ly in zona di memoria indirizzata da Mx (o Ax)
	LMIP	"	L(y)	EExy	Ly in zona di memoria indirizzata da Mx (o Ax) con incremento di 1 del contenuto di x
	LMIM	"	L(y)	EDxy	Ly in zona di memoria indirizzata da Mx (o Ax) con decremento di 1 del contenuto di x
	LPMIP	"	L(y)	E2xy	Incrementa di 1 Ly; il risultato viene inviato alla zona di memoria indirizzata da Mx (o Ax); incrementa di 1 il contenuto di x.
Microistruzioni con indirizzamento diretto della memoria	MAD	A(x)	C(y)(y)	3xyy	Deposita in Ax il contenuto della memoria all'indirizzo diretto Cyy
	AMD	A(x)	C(y)(y)	2xyy	Deposita in memoria all'indirizzo diretto Cyy, il contenuto di Ax
Microistruzioni a caricamento immediato	CRTA	A(x)	C(y)(y)	7xyy	Carica yy in Ax
	CRTB	B(x)	C(y)(y)	5xyy	Carica yy in Bx
Microistruzioni di salto condizionato	SAD0	D(x)	nome simbolico	6α(yy	Esegue il salto a yy se Dx = 0 } α é la maschera del deviatore
	SADI	D(x)	"	6α(yy	Esegue il salto a yy se Dx = 1 }
	SADE	nome simbolico		4▭xx	Esegue il salto a xx se ECOF0 = 1

SUDDIVISIONE DELLE MICROISTRUZIONI PER FUNZIONE

MICROISTRUZIONI CHE OPERANO TRA REGISTRI	NOME	OPERANDO 1	OPERANDO 2	Cod. Esad.	DESCRIZIONE
Microistruzioni aritmetiche e logiche	ADDA	A(x)	B(y)	96xy	Addiziona [Ax + By +D100] con risultato in Ax; posiziona i deviatori
	ADDB	A(x)	B(y)	A6xy	Addiziona [Ax + By + D100] con risultato in By; posiziona i deviatori
	ADD	A(x)	B(y)	86xy	Addiziona [Ax + By + D100] con risultato trascurato posiziona i deviatori
	SOTA	A(x)	B(y)	C6xy	Sottrazione: [Ax + $\overline{By}$ + D100] con risultato in Ax; posiziona i deviatori
	SOTB	A(x)	B(y)	D6xy	Sottrazione: [Ax + $\overline{By}$ + D100] con risultato in By; posiziona i deviatori
	SOT	A(x)	B(y)	B6xy	Sottrazione: [Ax + $\overline{By}$ + D100] con risultato trascurato; posiziona i deviatori
	ANDA	A(x)	B(y)	A7xy	AND tra Ax e By con risultato in Ax; posiziona D101 se risultato = 0
	ANDB	A(x)	B(y)	B7xy	AND tra Ax e By con risultato in By; posiziona D101 se risultato = 0
	AND	A(x)	B(y)	97xy	AND tra Ax e By con risultato trascurato; posiziona D101 se risultato = 0
	ORA	A(x)	B(y)	F6xy	OR tra Ax e By con risultato in Ax; posiziona D101 se risultato = 0
	ORB	A(x)	B(y)	87xy	OR tra Ax e By con risultato in By; posiziona D101 se risultato = 0
	OR	A(x)	B(y)	E6xy	OR tra Ax e By con risultato trascurato; posiziona D101 se risultato = 0
	OREA	A(x)	B(y)	D7xy	OR esclusivo tra Ax e By con risultato in Ax; posiziona D101 se risultato = 0
	OREB	A(x)	B(y)	E7xy	OR esclusivo tra Ax e By con risultato in By; posiziona D101 se risultato = 0
	ORE	A(x)	B(y)	C7xy	OR esclusivo tra Ax e By con risultato trascurato; posiziona D101 se risultato = 0
Microistruzioni di trasferimento	TAB	A(x)	B(y)	D8xy	Trasferisce Ax in By
	TBA	A(x)	B(y)	E9xy	Trasferisce By in Ax
	TABM	A(x)	B(y)	E8 xy	Trasferisce i 4 bit meno significativi di Ax in By
	TABP	A(x)	B(y)	D9xy	Trasferisce i 4 bit più significativi di Ax in By
	TBAM	A(x)	B(y)	F9xy	Trasferisce i 4 bit meno significativi di By in Ax
	TBAP	A(x)	B(y)	F8xy	Trasferisce i 4 bit più significativi di By in Ax
	SAB	A(x)	B(y)	BAxy	Scambio tra Ax e By
	SLL	L(x)	L(y)	BCxy	Scambio incrociato tra Lx e Ly

ELENCO DELLE MICROISTRUZIONI IN ORDINE DI CODICE ESADECIMALE

COD. ESAD	NOME	OPERANDO 1	OPERANDO 2
0XXX	SAI		
2XYY	AMD	A(X)	C(Y)(Y)
3XYY	MAD	A(X)	C(Y)(Y)
4XXX	SADE		
5XYY	CRTB	B(X)	C(Y)(Y)
6XYY	SADI	D(X)	
6XYY	SADD	D(X)	
7XYY	CRTA	A(X)	C(Y)(Y)
82XY	AMIM	A(0≤X≤B)	A(Y)
"	"	M(0≤X≤B)	A(Y)
83XF	TADI	A(X)	
85XF	ICA	A(X)	
86XY	ADD	A(X)	B(Y)
87XY	ORB	A(X)	B(Y)
88XY	AMIP	A(C≤X≤F)	A(Y)
"	"	M(0≤X≤B)	A(Y)
89XY	BMI	A(C≤X≤F)	B(Y)
"	"	M(0≤X≤B)	B(Y)
8AXY	BMIM	A(C≤X≤F)	B(Y)
"	"	M(0≤X≤B)	B(Y)
8BXF	ROTA	A(X)	
8CXY	BMIP	A(C≤X≤F)	B(Y)
"	"	M(0≤X≤B)	B(Y)
8DX8	EMI	A(C≤X≤F)	
"	"	M(0≤X≤B)	
8EXF	VRA	A(X)	
90X0	MEI	A(C≤X≤F)	
"	"	M(0≤X≤B)	
91XY	MAI	A(C≤X≤F)	A(Y)
"	"	M(0≤X≤B)	A(Y)
92XY	MAIM	A(C≤X≤F)	A(Y)
"	"	M(0≤X≤B)	A(Y)
93XF	TBDI	B(X)	
94X0	MEIP	A(C≤X≤F)	
"	"	M(0≤X≤B)	
95XF	ICB	B(X)	
96XY	ADDA	A(X)	B(Y)
97XY	AND	A(X)	B(Y)
98XY	MAIP	A(C≤X≤F)	A(Y)
"	"	M(0≤X≤B)	A(Y)
99XY	MBI	A(C≤X≤F)	B(Y)
"	"	M(0≤X≤B)	B(Y)
9AXY	MBIM	A(C≤X≤F)	B(Y)
"	"	M(0≤X≤B)	B(Y)
9BXF	ROTB	B(X)	
9CXY	MBIP	A(C≤X≤F)	B(Y)
"	"	M(0≤X≤B)	B(Y)
9DX0	MEIM	A(C≤X≤F)	
"	"	M(0≤X≤B)	
9EXF	VRB	B(X)	
A1X8	EMIM	A(C≤X≤F)	
"	"	M(0≤X≤B)	
A2X8	EMIP	A(C≤X≤F)	
"	"	M(0≤X≤B)	
A3XF	SDIA	A(X)	
A5XF	ICL	L(X)	
A6YX	ADDB	A(X)	B(Y)
A7XY	ANDA	A(X)	B(Y)
A8XY	AMI	A(C≤X≤F)	A(Y)
"	"	M(0≤X≤B)	A(Y)
A9X8	EDB	B(X)	
AAX0	ENTL	L(X)	
ABXF	AZAM	A(X)	
ADXF	EDC	L(X)	
AEXF	DCA	A(X)	
B1X4	ESE	A(C≤X≤F)	
"	"	M(0≤X≤B)	
B2XF	ETIB	B(X)	
B3XF	SDIB	B(X)	
B4X2	ECO	A(C≤X≤F)	
"	"	M(0≤X≤B)	
B6XY	SOT	A(X)	B(Y)
B7XY	ANDB	A(X)	B(Y)
B8X8	EDA	A(X)	
B9X0	ENUA	A(X)	
BAXY	SAB	A(X)	B(Y)
BBXF	AZAP	A(X)	
BCXY	SLL	L(X)	L(Y)
BD00	COM0		
BD10	COM1		
BD30	COM3		
BD40	COM4		
BD50	COM5		
BD60	COM6		
BD70	COM7		
BD80	COM8		
BD90	COM9		
BDA0	COM10		
BDB0	COM11		
BDC0	COM12		
BDD0	COM13		
BDE0	COM14		
BDF0	COM15		
BEXF	DCB	B(X)	
C3X0	SHDA	A(X)	
C3X1	SLDA	A(X)	
C4X0	SHSA	A(X)	
C4X1	SLSA	A(X)	
C5XF	TDIA	A(X)	
C6XY	SOTA	A(X)	B(Y)
C7XY	ORE	A(X)	B(Y)
C8XX	REDI	C(X)(X)	
C900	NOP		
C9XX	SEDI	C(X)(X)	
CAX0	TCCA	A(X)	
CBXF	AZBM	B(X)	
D1XY	MLI	A(C≤X≤F)	L(Y)
"	"	M(0≤X≤B)	L(Y)
D3X0	SHDB	B(X)	
D3X1	SLDB	B(X)	
D4X0	SHSB	B(X)	
D4X1	SLSB	B(X)	
D5XF	TDIB	B(X)	
D6XY	SOTB	A(X)	B(Y)
D7XY	OREA	A(X)	B(Y)
D8XY	TAB	A(X)	B(Y)
D9XY	TABP	A(X)	B(Y)
DAX1	TDMA	A(X)	
DBXF	AZAP	B(X)	
DDXY	MLIM	A(C≤X≤F)	L(Y)
"	"	M(0≤X≤B)	L(Y)
DEXY	MLIP	A(C≤X≤F)	L(Y)
"	"	M(0≤X≤B)	L(Y)
E0X8	ESI	A(C≤X≤F)	
"		M(0≤X≤B)	
E1XY	LMI	A(C≤X≤F)	L(Y)
"	"	M(0≤X≤B)	L(Y)
E2XY	LPMIP	A(C≤X≤F)	L(Y)
E2XY	LPMIP	M(0≤X≤B)	L(Y)
E5XF	DCL	L(X)	
E6XY	OR	A(X)	B(Y)
E7XY	OREB	A(X)	B(Y)
E8XY	TABM	A(X)	B(Y)
E9XY	TBA	A(X)	B(Y)
EAX2	TDPA	A(X)	
EBX8	ESIP	A(C≤X≤F)	
"	"	M(0≤X≤F)	
ECX8	ESIM	A(C≤X≤F)	
"	"	M(0≤X≤B)	
EDXY	LMIM	A(C≤X≤F)	L(Y)
"	"	M(0≤X≤B)	L(Y)
EEXY	LMIP	A(C≤X≤F)	L(Y)
"	"	M(0≤X≤B)	L(Y)
F1X0	SEI	A(C≤X≤F)	
"	"	M(0≤X≤B)	
F5XF	VRL	L(X)	
F6XY	ORA	A(X)	B(Y)
F7X0	SEIP	A(C X F)	
"	"	M(0 X B)	
F8XY	TBAP	A(X)	B(Y)
F9XY	TBAM	A(X)	B(Y)
FAXY	TABC	A(X)	B(Y)
FBX8	DEA	L(X)	
FCX0	DAE	L(X)	
FCX2	CAE	L(X)	
FDX0	SEIM	A(C≤X≤F)	
"	"	M(0≤X≤B)	

N	NOME	OPERANDO 1	OP. 2	CODICE ESAD.	REGIST. R0	DESCRIZIONE	DEVIATORI D1000	D1010	D1020	CROM	TROM	VROM B,A	FLUSSO DI FUNZIONAMENTO
	INCD0	D (x)	L (y)	A0∝y	A0∝y	Incrementa di 1 il contenuto di L(y) se il deviatore D (x) è a zero. Il valore di (x) è dato dai 3 bit più significativi della cifra ∝ . Il bit meno significativo di ∝ deve essere zero. Esempio: per un incremento condizionato da D105 occorre avere la cifra: ∝ = 1010 = A (1010: Deviatore D105)				C14E	1F	60	
	INCD1	D (x)	L (y)	A0∝y	A0∝y	Incrementa di 1 il contenuto di L (y) se il deviatore D (x) è a 1. Il valore di (x) è dato dai 3 bit più significativi della cifra ∝ . Il bit meno significativo di ∝ deve essere ad 1. Esempio: per un incremento condizionato da D105 occorre avere la cifra: ∝ = 1011 = B (1011: Deviatore D105)				C14B	1F	60	

	NOME	OPERANDO 1	OP. 2	COD. ESAD.	REGIST. R0	DESCRIZIONE	DEVIATORI D1000	DEVIATORI D1010	DEVIATORI D1020	CROM	TROM	VROM B,A	FLUSSO DI FUNZIONAMENTO
115	TBA	A(x)	B(y)	E9xy	E9xy	Trasferisce il contenuto di B(y) in A(x).				BFBF	13	18	RB → UB → NA → BA → RA
116	TBAM	A(x)	B(y)	F9xy	F9xy	Trasferisce i quattro bit meno significativi di B(y) nei quattro bit meno significativi di A(x), lasciando inalterati i quattro più significativi.				AB7F	11	18	RB → UB → NA → BA → RA
117	TBAP	A(x)	B(y)	F8xy	F8xy	Trasferisce i quattro bit più significativi di B(y) nei quattro bit più significativi di A(x) lasciando inalterati i quattro meno significativi.				AB7F	12	18	RB → UB → NA → BA → RA
118	TBDI	B(x)		93xF	93xF	Trasferisce il contenuto di B(x) sui deviatori.	1° BIT di B(x)	2° BIT di B(x)	3° BIT di B(x)	6BFF	50	04	RB → UA → DEV
119	TCCA	A(x)		CAx0	CAx0	Trasferisce i bit CONØ÷7 che rappresentano la condizione di console. da "SUCE 2" sul registro A(x) CON Ø Step by step CON 1 Start CON 2 Micro comando: codice operativo 0xx CON 3 " " " 8xx CON 4 " " " 5xx CON 5 Pulsante ABSOLUTE CON 6 Micro comando: codice operativo 6xx CON 7 " " " 7xx				BFED	13	06	SUCE 2 – CONO÷7 → BA → RA
120	TDIA	A(x)		C5xF	C5xF	Trasferisce il contenuto dei deviatori in A(x).				BEFF	13	06	DEV → NA → BA → RA
121	TDIB	B(x)		D5xF	D5xF	Trasferisce il contenuto dei deviatori su B(x)				7EFF	1C	06	DEV → NA → BB → RB
122	TDMA	A(x)		DAx1	DAx1	Trasferisce i bit CONØ÷7 da SUCE2 nel registro A(x); i bit CONO÷7 rappresentano gli 8 bit meno significativi del buffer di tastiera (BK00÷15) o del buffer di alt indirizzato(IC00÷15). Si preleva dal buffer di tastiera se la TDMA è preceduta da una COM12. da Reset o se si è passati da assetto microprogramma ad assetto programma ; si preleva dal buffer di alt indirizzato se si è impostato un alt indirizzato e se si è eseguita una COM14.				BFED	13	06	SUCE 2 – CONO÷7 → BA → RA
123	TDPA	A(x)		EAx2	EAx2	Trasferisce i bit CONØ÷7 (dati più significativi) da SUCE2 nel registro A(x); opera come la TDMA con l'unica differenza che vengono prelevati gli 8 bit più significativi del buffer di tastiera o di alt indirizzato.				BFED	13	06	SUCE 2 – CONO÷7 → BA → RA
124	VRA	A(x)		8ExF	8ExF	Pone ad uno D101 se il contenuto di A(x) è zero.		A(x)=0		3FF2	50	08	RA → UA → D101
125	VRB	B(x)		9ExF	9ExF	Pone ad uno D101 se il contenuto di B(x) è zero.		B(x)=0		6BF2	50	08	RB → UA → D101
126	VRL	L(x)		F5xF	F5xF	Pone ad uno D101 se il contenuto di L(x) è zero.		L(x)=0		3FF2	50	10	RA → UA → D101 RB → UB → D101

No.	NOME	OPERANDO 1	OP. 2	COD. ESAD.	REGIST. R0	DESCRIZIONE	DEVIATORI D1000	D1010	D1020	CROM	TROM	VROM B,A	FLUSSO DI FUNZIONAMENTO
104	SLL	L(x)	L(y)	BCxy	BCxy	Scambia il contenuto di L(x) ed L(y) incrociando gli 8 bit più significativi con gli 8 bit meno significativi, ovvero si scambiano rispettivamente A(x) con B(y) e A(y) con B(x).				BF9F	0F	18	RA→UA→NB→BB→RB RB→UB→NA→BA→RA β_1
										7F9F	1F	18	RA→UA→NB→BB→RB RB→UB→NA→BA→RA β_2
105	SLSA	A(x)		C4x1	C4x1	Shift lungo di A(x) a sinistra, ovvero tutto A(x) shifta a sinistra di un bit; il bit più significativo di A(x) va su D100 e il vecchio contenuto di D100 va sul bit meno significativo di A(x).	8° BIT di A(x)			BE71	53	18	RA → UA → NA → BA → RA ↳D100↑
106	SLSB	B(x)		D4x1	D4x1	Shift lungo a sinistra di B(x) ovvero tutto B(x) shifta a sinistra di un bit; il bit più significativo di B(x) va su D100 e il vecchio contenuto di D100 va sul bit meno significativo di B(x)	8° BIT di B(x)			6A71	5C	18	RB → UA → NA → BB → RB ↳D100↑
107	SOT	A(x)	B(y)	B6xy	B6xy	Esegue la sottrazione tra A(x) e B(y) in questo modo: [A(x) + $\overline{B(y)}$ + D100] ; D100 rappresenta l'eventuale riporto di un'operazione precedente. Posiziona i deviatori mentre il risultato non va in nessun registro.	>255	=0	>15	8DF7	50	08	RA ↘ RB → UA → DEV D100 ↗
108	SOTA	A(x)	B(y)	C6xy	C6xy	Esegue la sottrazione tra A(x) e B(y) in questo modo: [A(x) + $\overline{B(y)}$ + D100] ; D100 rappresenta l'eventuale riporto di un'operazione precedente. Il risultato viene posto in A(x) Posiziona i deviatori.	>255	=0	>15	8D77	53	18	RA ↘ RB → UA → NA → BA → RA D100 ↗ ↘DEV
109	SOTB	A(x)	B(y)	D6xy	D6xy	Esegue la sottrazione tra A(x) e B(y) in questo modo: [A(x) + $\overline{B(y)}$ + D100] ; D100 rappresenta l'eventuale riporto di un'operazione precedente. Il risultato viene posto in B(y). Posiziona i deviatori.	>255	=0	>15	8D97	5C	18	RA ↘ RB → UA → NB → BB → RB D100 ↗ ↘DEV
110	TAB	A(x)	B(y)	D8xy	D8xy	Trasferisce il contenuto di A(x) in B(y).				BF9F	1C	18	RA → UA → NB → BB → RB
111	TABC	A(x)	B(y)	FAxy	FAxy	Trasferisce il contenuto di A(x) e B(y) in "SUCE 2" accompagnato dal comando C010. VISUALIZZA				BF4C	10	18	RA → UA → NA → BA → SUCE 2 RB → UB → NB → BB →
112	TABM	A(x)	B(y)	E8xy	E8xy	Trasferisce i quattro bit meno significativi di A(x) nei quattro meno significativi di B(y) lasciando inalterati gli altri.				BF9F	14	18	RA → UA → NB → BB → RB
113	TABP	A(x)	B(y)	D9xy	D9xy	Trasferisce i quattro bit più significativi di A(x) nei quattro più significativi di B(y), lasciando inalterati gli altri.				BF9F	18	18	RA → UA → NB → BB → RB
114	TADI	A(x)		83xF	83xF	Trasferisce sui deviatori il contenuto del registro A (x).	1° BIT di A(x)	2° BIT di A(x)	3° BIT di A(x)	BFFF	50	04	RA → UA → DEV

N°	NOME	OPERANDO 1	OPER. 2	COD. ESAD.	REGIST. R0	DESCRIZIONE	DEVIATORI D1000	D1010	D1020	CROM	TROM	VROM B,A	FLUSSO DI FUNZIONAMENTO
94	SEDI	C(x)(x)		C9xx	C9XX	Setta i deviatori mascherati da C(x)(x) e lascia inalterati gli altri. Se la costante C (x)(x) della maschera fosse 21 verrebbero settati D100 e D105.				FFE7	50	01	R000 ÷ 7 → DEV
95	SEI	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$		F1x0	F1x0	Invia all'esterno su ECD0 ÷ ECDF la parola di memoria indirizzata indirettamente da A12≤(x)≤15 o da M0≤(x)≤11. Emette lo strobe "ECOT". L'indirizzo di memoria va inteso come indirizzo di parola; ad esempio, all'indirizzo 0002 corrisponde la parola 0002 cioè i byte 0004 e 0005.				3FFF BF0F	A0 10	0C 1F	RA, RB → //16 BC → MEMORIA β_1 ME00÷7 → BM → NA → BA ME08÷15 → NB → BB → ECD0÷F β_2
96	SEIM	" "		FDx0	FDx0	Invia all'esterno su ECD0 ÷ ECDF la parola di memoria indirizzata indirettamente da A12≤(x)≤15 o da M0≤(x)≤11. L'indirizzo è decrementato di uno. Emette lo strobe "ECOT". L'indirizzo di memoria va inteso come indirizzo di parola; ad esempio all'indirizzo 0002 corrisponde la parola 0002 cioè i byte 0004 e 0005.				1F5F BF0F	AF 10	60 7C	RA → -1 → UA → NA → BA → RA RB → UB → NB → BB → RB → //16 BC — MEMORIA β_1 ME00÷7 → BM → NA → BA ME08÷15 → NB → BB → ECD0÷F β_2
97	SEIP	" "		F7x0	F7x0	Invia all'esterno su ECD0 ÷ ECDF la parola di memoria indirizzata indirettamente da A12≤(x)≤15 o da M0≤(x)≤11. L'indirizzo è incrementato di uno. Emette lo strobe "ECOT". L'indirizzo di memoria va inteso come indirizzo di parola; ad esempio, all'indirizzo 0002 corrisponde la parola 0002 cioè i byte 0004 e 0005.				015F BF0F	AF 10	60 7C	RA → +1 → UA → NA → BA → RA RB → UB → NB → BB → RB → //16 BC → MEMORIA β_1 ME00÷7 → BM → NA → BA ME08÷15 → NB → BB → ECD0÷F β_2
98	SHDA	A(x)		C3x0	C3x0	Shift a destra di A ovvero A(x) shifta a destra di un bit; il bit meno significativo di A(x) va su D100 e il bit più significativo di A(x) è forzato a zero.	1° BIT di A(x)			BE31	53	18	RA → UA → D100 NA → BA → RA
99	SHDB	B(x)		D3x0	D3x0	Shift a destra di B; ovvero B(x) shifta a destra di un bit; il bit meno significativo di B(x) va su D100 e il bit più significativo di B(x) è forzato a zero.	1° BIT di B(x)			6A31	5C	18	RB → UA → D100 NA → BB → RB
100	SHSA	A(x)		C4x0	C4x0	Shift a sinistra di A; ovvero A(x) shifta a sinistra di un bit; nel bit meno significativo di A(x) è forzato uno zero ed il bit più significativo va su D100.	8° BIT di A(x)			BE71	53	18	RA → UA → D100 NA → BA → RA
101	SHSB	B(x)		D4x0	D4x0	Shift a sinistra di B; ovvero B(x) shifta a sinistra di un bit; il bit meno significativo di B(x) è forzato a zero ed il bit più significativo va su D100.	8° BIT di B(x)			6A71	5C	18	RB → UA → D100 NA → BB → RB
102	SLDA	A(x)		C3x1	C3x1	Shift lungo a destra di A(x) ovvero A(x) shifta a destra di un bit, il bit meno significativo va su D100 ed il vecchio contenuto di D100 va sul bit più significativo di A(x).	1° BIT di A(x)			BE31	53	18	RA → UA → NA → BA → RA ↳ D100 ↑
103	SLDB	B(x)		D3x1	D3x1	Shift lungo a destra di B(x); ovvero B(x) shifta a destra di un bit, il bit meno significativo di B(x) va su D100 ed il vecchio contenuto di D100 va sul bit più significativo diB(x)	1° BIT di B(x)			6A31	5C	18	RB → UA → NA → BB → RB ↳ D100 ↑

No	NOME	OPERANDO 1	OPER 2	COD. ESAD.	REGIST. R0	DESCRIZIONE	DEVIATORI D1000	D1010	D1020	CROM	TROM	VROM B,A	FLUSSO DI FUNZIONAMENTO
88	SADE	Nome simbolico		4⊠xx	CF⊠Z	Salto condizionato dal deviatore esterno ECOF; il salto avviene se ECOF0 = 1. Il salto consiste nel portare xx nel registro indirizzatore del microprogramma in corso. L'operando è un nome simbolico che indica in quale punto del microprogramma si effettua il salto.				FF2B	13	06	ME00÷7→BM→NA→BA→RA (FASE α / FASE β)
89	SADI	D(x)	Nome simbolico	6αYY	EFαZ	Salto condizionato dal deviatore D(x); il salto avviene se D(x) è ad 1. Il valore di (x) è dato dai 3 bit più significativi della cifra α. Il bit meno significativo di α deve essere ad 1. Ad esempio: per un salto condizionato da D105 occorre avere la cifra: α = 1011 = B (1 → Deviatore D105). Il secondo operando è un nome simbolico che indica in quale punto del microprogramma si effettua il salto. Il salto consiste nel portare le cifre yy nel registro indirizzatore di microprogramma del livello in corso.				FF2B	13	06	ME00÷7→BM→NA→BA→RA (FASE α / FASE β)
90	SADO	D(x)	Nome simbolico	6αYY	EFαZ	Salto condizionato dal deviatore D(x); il salto avviene se D(x) è a zero. Il valore di (x) è dato dai 3 bit più significativi della cifra α. Il bit meno significativo di α deve esser zero. Esempio: per un salto condizionato da D105 occorre avere la cifra α = 1010 = A (1 → Deviatore D105). Il secondo operando è un nome simbolico che indica in quale punto del microprogramma si salta. Il salto consiste nel portare le cifre yy nel registro indirizzatore di microprogramma del livello in corso.				FF2B	13	06	ME00÷7→BM→NA→BA→RA (FASE α / FASE β)
91	SAI	Nome simbolico		0xxx (13 BIT)	8 / 9 ⊠Z (*)	Salto incondizionato. Il nuovo indirizzo è indicato dai 13 bit meno significativi del codice esadecimale. Il salto consiste nel trasferire i 13 bit nel registro indirizzatore di microprogramma per i livelli 4 e 3 (LØØ e LØ1). I tre bit più significativi di LØØ o LØ1 non vengono alterati. Per i livelli 2 e 1 i registri indirizzatori sono rispettivamente A13 e A12; in questo caso si modifica solo A12 e A13 senza alterare B12 e B13.				FF09	1F	06	FASE α / FASE β: ME00÷7→BM→NA→BA→RA; ME08÷12, RB05÷7 → NB→BB→RB
92	SDIA	A(x)		A3xF	A3xF	Scambia il contenuto dei deviatori D1000 ÷ 7 con il contenuto di A(x).				BEFF	53	0C	RA→UA→DEV; NA→BA→RA
93	SDIB	B(x)		B3xF	B3xF	Scambia il contenuto dei deviatori D1000 ÷ 7 con il contenuto di B(x).				6AFF	5C	0C	RB→UA→DEV; NA→BB→RB

(*) Il 13° bit del registro R0 può essere 0 o 1 a seconda del valore di xxx nel codice esadecimale della microistruzione; si possono selezionare perciò due posizioni di nanoro m che contengono però lo stesso codice.

No.	NOME	OPERANDO 1	OPER 2	COD. ESAD.	REGIST. R0	DESCRIZIONE	DEVIATORI D1000	DEVIATORI D1010	DEVIATORI D1020	CROM	TROM	VROM B,A	FLUSSO DI FUNZIONAMENTO
76	NOP			C900	C900	Non opera, non altera lo scratch-pad nè i deviatori. Viene incrementato di uno il registro indirizzatore del microprogramma in esecuzione: LØØ oppure LØ1, oppure A12, oppure A13.				FFE7	50	01	Il registro indirizzatore di microprogramma è incrementato in fase α.
77	OR	A(x)	B(y)	E6xy	E6xy	Esegue l'OR tra i contenuti di A(x) e B(y). Se il risultato è zero posiziona a 1 il deviatore D101.		=0		AFF2	50	08	RA, RB → UA → D101
78	ORA	A(x)	B(y)	F6xy	F6xy	Esegue l'OR tra i contenuti di A(x) e B(y). Porta il risultato in A(x) e posiziona il deviatore D101.		=0		AF72	53	18	RA, RB → UA → D101; NA-BA-RA
79	ORB	A(x)	B(y)	87xy	87xy	Esegue l'OR tra i contenuti di A(x) e B(y). Porta il risultato in B(y) e posiziona il deviatore D101.		=0		AF72	5C	18	RA, RB → UA → D101; NB-BB-RB
80	ORE	A(x)	B(y)	C7xy	C7xy	Esegue l'OR esclusivo tra i contenuti di A(x) e B(y), e posiziona il deviatore D101. Riportiamo la tabella della verità della funzione OR esclusivo ⊕: 1⊕1 =0; 0⊕0 =0; 1⊕0 =1; 0⊕1 =1		=0		ADF2	50	08	RA, RB → UA → D101
81	OREA	A(x)	B(y)	D7xy	D7xy	Deposita in A(x) il risultato dell'operazione OR esclusivo tra il contenuto di A(x) e B(y) e posiziona il deviatore D101. Riportiamo la tabella della verità della funzione OR esclusivo ⊕: 1⊕1 =0; 0⊕0 = 0; 1⊕0 =1; 0⊕1 =1		=0		AD72	53	18	RA, RB → UA → D101; NA→BA→RA
82	OREB	A(x)	B(y)	E7xy	E7xy	Deposita in B(y) il risultato dell'operazione OR esclusivo tra il contenuto dei registri A(x) e B(y). Posiziona il deviatore D101. Riportiamo la tabella della verità della funzione OR esclusivo ⊕: 1⊕1 =0; 0⊕0 =0; 1⊕0 =1; 0⊕1 =1		=0		AD92	5C	18	RA, RB → UA → D101; NB→BB→RB
83	REDI	C(x)(x)		C8xx	C8xx	Resetta i deviatori mascherati da C(x)(x) e lascia inalterati gli altri; se la costante C della maschera fosse 21 (codice esadecimale) verrebbero resettati i deviatori D100 e D105.				FFE7	50	01	R000÷7 → DEV
84	RESE	REGISTRO INDIRIZZATORE DI MICROPROGRAMMA DI LIVELLO 3 LØ1			50 ⊠ LØ1	La RESE è una pseudo-microistruzione che viene eseguita con il Reset; il registro R0 non viene caricato da un codice di memoria ma direttamente dall'hardware di Unità Centrale. Simula le condizioni di fine fase α e carica in LØ1 la 1ª μ-istruzione di inizializzazione (indir. esad: 8ØØØ)				F89F	1F	30	UA → NA→BA→RA; NB→BB→RB
85	ROTA	A(x)		8BxF	8BxF	Ruota la parte più significativa di A(x) con la parte meno significativa, ovvero scambia RA00 con RA04, RA01 con RA05, RA02 con RA06, RA03 con RA07.				BEBF	13	18	RA→UA→NA→BA→RA
86	ROTB	B(x)		9BxF	9BxF	Ruota la parte più significativa di B(x) con la parte meno significativa, ovvero scambia RB00 con RB04, RB01 con RB05, RB02 con RB06, RB03 con RB07.				6ABF	1C	18	RB→UA→NA→BB→RB
87	SAB	A(x)	B(y)	BAxy	BAxy	Scambia il contenuto di A(x) con il contenuto di B(y).				BF9F	1F	30	RA→UA→NB→BB→RB; RB→UB→NA→BA→RA

N°	NOME	OPERANDO 1	OP. 2	COD. ESAD.	REGIST. R0	DESCRIZIONE	DEVIATORI D1000	DEVIATORI D1010	DEVIATORI D1020	CROM	TROM	VROM B,A	FLUSSO DI FUNZIONAMENTO
67	MBI	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$	B(y)	99xy	99xy	Deposita in B(y) il contenuto del byte di memoria indirizzato indirettamente da M0≤(x)≤11 o da A12≤(x)≤15.				3FFF	A0	CC	RA, RB → BC (1/8) → MEMORIA β_1
										FF3F	1C	00	ME00÷7 oppure ME08÷15 → BM→NA→BB→RB β_2
68	MBIM	"	B(y)	9Axy	9Axy	Deposita in B(y) il contenuto del byte di memoria indirizzato indirettamente da M0≤(x)≤11 o da A12≤(x)≤15; decrementa di uno l'indirizzo di memoria.				1F5F	AF	60	RA → UA (-1) → NA→BA→RA RB → UB → NB→BB→RB BC (1/8) → MEMORIA β_1
		"								FF3F	1C	06	ME00÷7 oppure ME08÷15 → BM→NA→BB→RB β_2
69	MBIP	"	B(y)	9Cxy	9Cxy	Deposita in B(y) il contenuto del byte di memoria indirizzato indirettamente da M0≤(x)≤11 o da A12≤(x)≤15; incrementa di uno l'indirizzo di memoria.				015F	AF	60	RA → UA (+1) → NA→BA→RA RB → UB → NB→BB→RB BC (1/8) → MEMORIA β_1
		"								FF3F	1C	06	ME00÷7 oppure ME08÷15 → BM→NA→BB→RB β_2
70	MEI	"		90x0	90x0	Invia su ECD0 ÷ ECD7 verso esterno, il byte di memoria indirizzato indirettamente da M0≤(x)≤11 o da A12≤(x)≤15.				3FFF	A0	0C	RA, RB → BC (1/8) → MEMORIA β_1
		"								FF2F	10	3E	ME00÷7 oppure ME08÷15 → BM→NA→BA→ECD0÷7 β_2
71	MEIM	"		9Dx0	9Dx0	Invia su ECD0 ÷ ECD7 verso esterno, il byte di memoria indirizzato indirettamente da M0≤(x)≤11 o da A12≤(x)≤15 e decrementa di uno l'indirizzo.				1F5F	AF	60	RA → UA (-1) → NA→BA→RA RB → UB → NB→BB→RB BC (1/8) → MEMORIA β_1
		"								FF2F	10	7C	ME00÷7 oppure ME08÷7 → BM→NA→BA→ECD0÷7 β_2
72	MEIP	"		94x0	94x0	Invia su ECD0 ÷ ECD7 verso esterno il byte di memoria indirizzato indirettamente da M0≤(x)≤11 o da A12≤(x)≤15 ed incrementa di uno l'indirizzo.				015F	AF	60	RA → UA (+1) → NA→BA→RA RB → UB → NB→BB→RB BC → MEMORIA β_1
		"								FF2F	10	7C	ME00÷7 oppure ME08÷15 → BM→NA→BA→ECD0÷7 β_2
73	MLI	"	L(y)	D1xy	D1xy	Deposita in L(y) il contenuto della memoria indirizzata indirettamente da M0≤(x)≤11 o da A12≤(x)≤15. L'indirizzo va inteso come indirizzo di parole; esempio: l'indirizzo esadecimale 0002 corrisponde alla parola 0002 ovvero ai byte 0004 e 0005.				3FFF	A0	CC	RA, RB → BC (1/16) → MEMORIA β_1
		"								FF1F	1F	00	ME00÷7→BM→NA→BA→RA ME08÷15→NB→BB→RB β_2
74	MLIM	"	L(y)	DDxy	DDxy	Deposita in L(y) il contenuto della memoria indirizzata indirettamente da M0≤(x)≤11 o da A12≤(x)≤15, decrementa di uno l'indirizzo di memoria. L'indirizzo va inteso come indirizzo di parole; esempio: l'indirizzo di memoria esadecimale 0002 corrisponde alla parola 0002 ovvero ai byte 0004 e 0005.				1F5F	AF	60	RA → UA (-1) → NA→BA→RA RB → UB → NB→BB→RB BC (1/16) → MEMORIA β_1
		"								FF1F	1F	06	ME00÷7→BM→NA→BA→RA ME08÷15→NB→BB→RB β_2
75	MLIP	"	L(y)	DExy	DExy	Deposita il L(y) il contenuto della memoria indirizzata indirettamente da M0≤(x)≤11 o da A12≤(x)≤15 ed incrementa di uno l'indirizzo di memoria. L'indirizzo di memoria va inteso come indirizzo di parole; esempio: all'indirizzo esadecimale di memoria 0002 corrispondono i byte 0004 e 0005.				015F	AF	60	RA → UA (+1) → NA→BA→RA RB → UB → UB→BB→RB BC (1/16) → MEMORIA β_1
		"								FF1F	1F	06	ME00÷7→BM→NA→BA→RA ME08÷15→NB→BB→RB β_2

N°	NOME	OPERANDO 1	OP. 2	COD. ESAD.	REGIST. R0	DESCRIZIONE	DEVIATORI D1000	DEVIATORI D1010	DEVIATORI D1020	CROM	TROM	VROM B, A	FLUSSO DI FUNZIONAMENTO
58	ICL	L(x)		A5xF	A5xF	Incrementa di uno il contenuto di L(x)				015F	1F	60	RA → UA (+1) → NA → BA → RA RB → UB → NB → BB → RB
59	LMI	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$	L(y)	E1xy	E1xy	Scrive il contenuto di L(y) in memoria nella posizione indirizzata indirettamente da $M0 \leq (x) \leq 11$ o da $A12 \leq (x) \leq 15$. L'indirizzo va inteso come indirizzo di una parola; esempio: l'indirizzo esadecimale 0002 di parola, corrisponde ai byte 0004 e 0005.				3FFF	A0	2C	RA, RB → 11/16 BC → MEMORIA β_1
										FF5F	10	00	RA → UA → NA → BA → ECD0÷7 RB → UB → NB → BB → ECD8÷F } MEM. β_2
60	LMIM	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$	L(y)	EDxy	EDxy	Scrive il contenuto di L(y) in memoria nella posizione indirizzata indirettamente da $M0 \leq (x)$ 11 o da $A12 \leq (x) \leq 15$ e decrementa di uno l'indirizzo. L'indirizzo va inteso come indirizzo di una parola; esempio: l'indirizzo esadecimale 0002 di parola corrisponde ai byte 0004 e 0005.				1F5F	AF	60	RA → UA (-1) → NA → BA → RA RB → UB → NB → BB → RB β_1 BC (11/16) → MEMORIA
										FF5F	10	01	RA → UA → NA → BA → ECD0÷7 RB → UB → NB → BB → ECD8÷F } MEM. β_2
61	LMIP	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$	L(y)	EExy	EExy	Scrive il contenuto di L(y) in memoria nella posizione indirizzata indirettamente da $M0 \leq (x) \leq 11$ o da $A12 \leq (x) \leq 15$ e incrementa di uno l'indirizzo. L'indirizzo va inteso come indirizzo di una parola; esempio: l'indirizzo esadecimale 0002 di parola corrisponde ai byte 0004 e 0005.				015F	AF	60	RA → UA (+1) → NA → BA → RA RB → UB → NB → BB → RB β_1 BC (11/16) → MEMORIA
										FF5F	10	01	RA → UA → NA → BA → ECD0÷7 RB → UB → NB → BB → ECD8÷F } MEM. β_2
62	LPMIP	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$	L(y)	E2xy	E2xy	Indirizza indirettamente la memoria con $M0 \leq (x) \leq 11$ o con $A12 \leq (x) \leq 15$. Dopo l'indirizzamento M o A viene incrementato di 1. Il contenuto di Ly+1 viene scritto in memoria. L'indirizzo va inteso come indirizzo di una parola ; esempio: l'indirizzo esadecimale 0002 corrisponde ai byte 0004 e 0005.				015F	AF	60	RA (+1) → UA → NA → BA → RA RB → UB → NB → BB → RB β_1 BC (11/16) → MEMORIA
										C15F	10	01	RA (-1) → UA → NA → BA → ECD0÷7 RB → UB → NB → BB → ECD8÷F } MEM. β_2
63	MAD	A(x)	C(y)(y)	3xyy	BFX⊠	Deposita in A(x) il contenuto della memoria all'indirizzo diretto C(y)(y).				FF2E	20	66	FASE 4 ME00÷7 → BM → NA → BC (11/8) → MEM. β_1 RB = 0
										BF2E	13	00	ME00÷7 → BM → NA → BA → RA β_2
64	MAI	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$	A(y)	91xy	91xy	Deposita in A(y) il contenuto del byte di memoria indirizzato indirettamente da $M0 \leq (x) \leq 11$ o da $A12 \leq (x) \leq 15$.				3F5F	A0	CC	RA, RB → 11/8 BC → MEMORIA β_1
										FF3F	13	00	ME00÷7 oppure ME08÷15 → BM → NA → BA → RA β_2
65	MAIM	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$	A(y)	92xy	92xy	Deposita in A(y) il contenuto del byte di memoria indirizzato indirettamente da $M0 \leq (x) \leq 11$ o da $A12 \leq (x) \leq 15$. Decrementa l'indirizzo di memoria di uno.				1F5F	AF	60	RA → UA (-1) → NA → BA → RA RB → UB → NB → BB → RB β_1 BC (11/8) → MEMORIA
										FF3F	13	06	ME00÷7 oppure ME08÷15 → BM → NA → BA → RA β_2
66	MAIP	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$	A(y)	98xy	98xy	Deposita in A(x) il contenuto del byte di memoria indirizzato indirettamente da $M0 \leq (x) \leq 11$ o da $A12 \leq (x) \leq 15$. Incrementa l'indirizzo di memoria di uno.				015F	AF	60	RA → UA (+1) → NA → BA → RA RB → UB → NB → BB → RB β_1 BC (11/8) → MEMORIA
										FF3F	13	06	ME00÷7 oppure ME08÷15 → BM → NA → BA → RA β_2

No	NOME	OPERANDO 1	OP. 2	COD. ESAD.	REGIST. R0	DESCRIZIONE	DEVIATORI D1000	D1010	D1020	CROM	TROM	VROM B, A	FLUSSO DI FUNZIONAMENTO
48	EMIP	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$		A2x8	A2x8	Scrive in memoria, nel byte indirizzato indirettamente da $M0 \leq (x) \leq 11$ o da $A12 \leq (x) \leq 15$, il byte in arrivo da periferica su EPD0 ÷ EPD7. Incrementa di 1 il contenuto del registro indice.				015F	AF	60	RA → +1 → UA → NA → BA → RA RB → UB → NB → BB → RB → BC (11/8) → MEMORIA — β_1
										FFEF	10	1F	EPD0 ÷ 7 → NA → BA → ECD → MEMORIA — β_2
49	ENTL	L(x)		AAx0	AAx0	Introduce il bus nome (EPN0 ÷ EPN7) nella parte meno significativa di L(x) ovvero in A(x) ed il bus tipo (EPT0÷EPT7) nella parte più significativa di L(x), ovvero in Bx.				3FDF	1F	06	EPN0 ÷ 7 → NA → BA → RA EPT0 ÷ 7 → NB → BB → RB
50	ENUA	A(x)		B9x0	B9x0	Introduce il bus nome (EPN0 ÷ EPN7) in A(x); se nessun governo è selezionato viene introdotto il nome dell'UC. Non viene emesso ECOT				BFFF	13	06	EPN0 ÷ 7 → NA → BA → RA
51	ESE	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$		B1x4	B1x4	Emette la selezione: ovvero emette su ECD0÷ ECD7 il contenuto della memoria indirizzata indirettamente da $M0 \leq (x) \leq 11$ o da $A12 \leq (x) \leq 15$ accompagnato dagli strobe "ECOT" ed "ECOS".				3FFF	A0	0C	RA, RB → BC (11/8) → MEMORIA — β_1
										FF2F	10	7C	ME00 ÷ 7 (oppure ME08 ÷ 15) → BM → NA → BA → ECD0 ÷ 7 — β_2 vengono emessi ECOS e ECOT
52	ESI	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$		E0x8	E0x8	Scrive in memoria la parola in arrivo da UP (EPD0 ÷ 7, EPT0 ÷ 7) nella posizione di memoria indirizzata indirettamente da $M0 \leq (x) \leq 11$ o da $A12 \leq (x) \leq 15$. Emette inoltre lo strobe "ECOT". L'indirizzo va inteso come indirizzo di parole; supponiamo che il contenuto di M con $0 \leq (x) \leq 11$ sia, in codice esadecimale, 0002. In questo caso la parola selezionata è appunto la 0002 e i byte selezionati 0004 e0005				3FFF	A0	18	RA, RB → BC (11/6) → MEMORIA — β_1
										FFCF	10	1F	EPD0 ÷ 7 → NA → BA → ECD0 ÷ 7 EPT0 ÷ 7 → NB → BB → ECD8 ÷ F } MEM. — β_2 Viene emesso "ECOT"
53	ESIM	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$		ECx8	ECx8	Scrive in memoria la parola in arrivo da UP (EPD0 ÷ 7, EPT0 ÷ 7) nella posizione di memoria indirizzata indirettamente da $M0 \leq (x) \leq 11$ o da $A12 \leq (x) \leq 15$ e decrementa l'indirizzo di uno. Emette "ECOT". L'indirizzo va inteso come indirizzo di parole; supponiamo che il contenuto di M con $0 \leq (x) \leq 11$ sia, in codice esadecimale, 0002. In questo caso la parola selezionata è appunto la 0002 e i byte 0004 e 0005.				1F5F	AF	60	RA → -1 → UA → NA → BA → RA RB → UB → NB → BB → RB → BC (11/6) → MEMORIA — β_1
										FFCF	10	1F	EPD0 ÷ 7 → NA → BA → ECD0 ÷ 7 EPT0 ÷ 7 → NB → BB → ECD8 ÷ F } MEM. — β_2 Viene emesso "ECOT"
54	ESIP	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$		EBx8	EBx8	Scrive in memoria la parola in arrivo da UP (EPD0 ÷ 7, EPT0 ÷ 7) nella posizione di memoria indirizzata indirettamente da $M0 \leq (x) \leq 11$ o da $A12 \leq (x) \leq 15$ e incrementa l'indirizzo di uno. Emette "ECOT". L'indirizzo va inteso come indirizzo di parole; supponiamo che il contenuto di M con $0 \leq (x) \leq 15$ sia, in codice esadecimale, 0002. In questo caso la parola selezionata è appunto la 0002 e i byte 0004 e 0005.				015F	AF	60	RA → +1 → UA → NA → BA → RA RB → UB → NB → BB → RB → BC (11/6) → MEMORIA — β_1
										FFCF	10	1F	EPD0 ÷ 7 → NA → BA → ECD0 ÷ 7 EPT0 ÷ 7 → NB → BB → ECD8 ÷ F } MEM. — β_2 Viene emesso "ECOT"
55	ETIB	B(x)		B2xF	B2xF	Deposita il contenuto del canale tipo(EPT0÷7) nel registro B(x); non emette lo strobe "ECOT"				7FDF	1C	06	EPT0 ÷ 7 → NB → BB → RB
56	ICA	A(x)		85xF	85xF	Incrementa di uno il contenuto di A(x)				817F	13	18	RA → +1 → UA → NA → BA → RA
57	ICB	B(x)		95xF	95xF	Incrementa di uno il contenuto di B(x)				5B5F	1C	30	RB → +1 → UB → NB → BB → RB

N°	NOME	OPERANDO 1	OPER 2	COD. ESAD.	REGIST. R0	DESCRIZIONE	DEVIATORI D1000	DEVIATORI D1010	DEVIATORI D1020	CROM	TROM	VROM B, A	FLUSSO DI FUNZIONAMENTO
37	DAE	L(x)		FCx0	FCx0	Emissione dati verso periferiche: viene emesso su ECD0 + ECDF il contenuto del registro L(x) accompagnato dallo strobe di temporizzazione "ECOT" .				3F5F	00	18	RA→UA→NA→BA→ECDO÷7 RB→UB→NB→BB→ECD8÷F β_1
										3F4F	10	7C	RA→UA→NA→BA→ECDO÷7 RB→UB→NB→BB→ECD8÷F Viene emesso ECOT β_2
38	DCA	A(x)		AExF	AExF	Decrementa di uno il contenuto di A(x) e posiziona il deviatore D101:-a 1 se il risultato = 0 -a 0 negli altri casi		=0		9F72	53	30	RA→UA (-1) → D101 NA→BA→RA
39	DCB	B(x)		BExF	BExF	Decrementa di uno il contenuto di B(x) e posiziona il deviatore D101: -a 1 se il risultato = 0 -a 0 negli altri casi		=0		5D52	0C	30	RB→UB (-1) → NB→BB→RB β_1
										6BF2	50	04	RB→UA→D101 β_2
40	DCL	L(x)		E5xF	E5xF	Decrementa di uno il registro L(x) e posiziona il deviatore D101:-a 1 se il risultato = 0 -a 0 negli altri casi		=0		1F52	5F	C0	RA→UA (-1) → NA → D101 BA→RA RB→UB→NB→BB→RB
41	DEA	L(x)		FBx8	FBx8	Trasferisce verso esterno su ECD8 + F il contenuto della parte più significativa di L(x), ovvero il contenuto di B(x), ed introduce nella parte meno significativa di L(x), ovvero A(x), il contenuto di EPD0÷7. Viene emesso lo strobe "ECOT"				3F4F	00	78	RB→UB→NB→BB→ECDB÷F Viene emesso ECOT β_1
										3FDF	13	30	EPDO÷7→NA→BA→RA β_2
42	ECO	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$		B4x2	B4x2	Trasferisce il contenuto della memoria nelle posizioni indirizzate indirettamente da M0≤(x)≤11 o da A12≤(x)≤15, verso UP su ECD0 ÷ 7. Vengono emessi anche "ECOT"ed "ECOC" per dare il significato di comando al byte emesso.				3FFF	A0	0C	RA, RB → 11B BC → MEMORIA β_1
										FF2F	10	7C	MEO0÷7 oppure MEO8÷15 → BM→NA→BA→ECDO÷7 β_2 Vengono emessi ECOC e ECOT
43	EDA	A(x)		B8x8	B8x8	Deposita il contenuto del canale dati EPD0÷7 nel registro A(x); non viene emesso lo strobe "ECOT".				BFFF	13	06	EPDO÷7→NA→BA→RA
44	EDB	B(x)		A9x8	A9x8	Deposita il contenuto del canale dati(EPD0÷7) nel registro B(x); non viene emesso lo strobe "ECOT".				7FFF	1C	06	EPDO÷7→NA→BB→RB
45	EDC	L(x)		ADxF	ADxF	Decrementa di uno il contenuto dei 12 bit meno significativi di L(x); quando tutti i 12 bit sono a zero viene emesso ECOF				1F48	17	60	RA→UA (-1) → NA→BA→RA RB→UB→NB→BB→RB
46	EMI	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$		BDx8	BDx8	Scrive in memoria, nel byte indirizzato indirettamente da M0≤(x)≤11 o da A12≤(x)≤15, il byte in arrivo da periferica su EPD0 + 7.				3FFF	A0	0C	RA, RB → 11B BC → MEMORIA β_1
										FFEF	10	1F	EPDO÷7→NA→BA→ECD→MEMORIA β_2
47	EMIM	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$		A1x8	A1x8	Scrive in memoria, nel byte indirizzato indirettamente da M0≤(x)≤11 o da A12≤(x)≤15, il byte in arrivo da periferica su EPD0 + 7; decrementa di 1 il contenuto del registro indice.				1F5F	AF	60	RA→UA (-1) → NA→BA→RA RB→UB→NB→BB→RB 11B → BC → MEMORIA β_1
										FFEF	10	1F	EPDO÷7→NA→BA→ECD→MEMORIA β_2

No	NOME	OPERANDO 1	OP. 2	COD. ESAD.	REGIST. R0	DESCRIZIONE	DEVIATORI D1000	DEVIATORI D1010	DEVIATORI D1020	CROM	TROM	VROM A,B	FLUSSO DI FUNZIONAMENTO
26	COM7			BD70	BD70	Emette CE17N Setta "EC1F"				FFEA	10	1F	Non vengono interessati i blocchi; viene emesso solo il comando.
27	COM8			BD80	BD80	Emette CE18N Resetta "EC1F"				"	"	"	"
28	COM9			BD90	BD90	Emette CE19N Setta "EC2F"				"	"	"	"
29	COM10			BDA0	BDA0	Emette CE20N Resetta "EC2F"				"	"	"	"
30	COM11			BDB0	BDB0	Emette CE21N; nel "SUCE 2" oscura i display				"	"	"	"
31	COM12			BDC0	BDC0	Emette CE22N ; nel "SUCE 2" resetta il bit di START ; predispone il prelievo dei dati da buffer di tastiera (BK00 ÷ 15) del SUCE 2 per le successive TDMA o TDPA				"	"	"	"
32	COM13			BDD0	BDD0	Emette CE23N; nel "SUCE 2" segnala lo STOP su ALT INDIRIZZATO				"	"	"	"
33	COM14			BDE0	BDE0	Emette CE24N; predispone un indirizzo di ALT da "SUCE2" predispone il prelievo dei dati dal buffer di alt indirizzato (IC00 ÷ 15) del SUCE2 per le successive TDMA o TDPA				"	"	"	"
34	COM15			BDF0	BDF0	Emette CE25N; predispone il "SUCE 2" a visualizzare sul display "data" il dato emesso da U.C con una microistruzione TABC				"	"	"	"
35	CRTA	A(x)	C(YXY) o NOME SIMBOLICO	7xYY	FF x ☒	Trasferisce gli 8 bit yy del codice esadecimale nel registro A(x). L'operando 2 è chiamato "C" se il codice caricato in A(x) é un dato, è chiamato con un nome simbolico di una routine se è un indirizzo.				BF3F	13	06	ME00 ÷ 7 → BM → NA → BA → RA Fase α Fase β
36	CRTB	B(x)	C(YXY) o NOME SIMBOLICO	5xYY	DF x ☒	Trasferisce gli 8 bit yy del codice esadecimale nel registro B(x). L'operando 2 è chiamato "C" se il codice esadecimale trasferito in B(x) è un dato, è chiamato con un nome simbolico di una routine se è un indirizzo.				7F3F	1C	06	ME00 ÷ 7 → BM → NA → BB → RB Fase α Fase β

N°	NOME	OPERANDO 1	OP. 2	COD. ESAD.	REGIST. R0	DESCRIZIONE	DEVIATORI D1000	D1010	D1020	CROM	TROM	VROM B,A	FLUSSO DI FUNZIONAMENTO
16	BMI	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$	B(y)	89xy	89xy	Trasferisce in memoria il contenuto del registro B(y) all'indirizzo indiretto contenuto nel registro A12≤(x)≤15 o M0≤(x)≤11				3FFF	A0	2C	RA, RB → BC (118) → MEMORIA β1
										BFBF	10	00	RB → UB → NA → BA, BB } ECD → MEMORIA β2
17	BMIM	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$	B(y)	8Axy	8Axy	Trasferisce in memoria il contenuto del registro B(y) all'indirizzo indiretto contenuto nel registro A12≤(x)≤15 o M0≤(x)≤11. L'indirizzo contenuto in A12≤(x)≤15 o M0≤(x)≤11 è decrementato di uno.				1F5F	AF	60	RA → UA (-1) → NA → BA → RA RB → UB → NB → BB → RB BC (118) → MEMORIA β1
										FFBF	10	01	RB → UB → NA → BA, BB } ECD → MEMORIA β2
18	BMIP	$M_0 \leq (x) \leq 11$ $A_{12} \leq (x) \leq 15$	B(y)	8Cxy	8Cxy	Trasferisce in memoria il contenuto del registro B(y) all'indirizzo indiretto contenuto nel registro M0≤(x)≤11 o A12≤(x)≤15 L'indirizzo contenuto in M0≤(x)≤11 o in A12≤(x)≤15 è incrementato di uno				015F	AF	60	RA → UA (+1) → NA → BA → RA RB → UB → NB → BB → RB BC (118) → MEMORIA β1
										FFBF	10	01	RB → UB → NA → BA, BB } ECD → MEMORIA β2
19	CAE	L(x)		FCx2	FCx2	Emette verso esterno la riga L(x) con temporizzazioni di "ECOT" ed "ECOC"; la riga emessa ha il significato di comando per la unità periferica				3F5F	00	18	RA → UA → NA → BA → ECD0÷7 RB → UB → NB → BB → ECD8÷F β1
										3F4F	10	7C	RA → UA → NA → BA → ECD0÷7 RB → UB → NB → BB → ECD8÷F β2 Vengono emessi ECOC e ECOT
20	COM0			BD00	BD00	Emette CE10N. Disattiva il livello di lavoro in cui é emesso. Pone a 1 ECORN. Non viene emesso a liv. 4				FFEA	10	1F	Non vengono interessati i blocchi; si emette solo il comando
21	COM1			BD10	BD10	Emette CE11N. Attiva il microsupervisore e quindi il livello 3. Non ha effetto se emesso a livello 3.				"	"	"	"
22	COM3			BD30	BD30	Emette CE13N. Pone a 0 ECORN; si resettano tutti i governi periferici.				"	"	"	"
23	COM4			BD40	BD40	Emette CE14N. Fa nascere ECOT				"	"	"	"
24	COM5			BD50	BD50	Emette CE15N				"	"	"	"
25	COM6			BD60	BD60	Emette CE16N				"	"	"	"

No	NOME	OPERANDO 1	OP. 2	COD. ESAD.	REGIST. R0	DESCRIZIONE	DEVIATORI D1000	DEVIATORI D1010	DEVIATORI D1020	CROM	TROM	VROM B A	FLUSSO DI FUNZIONAMENTO
5	AMD	A(x)	C(y)(y)	2XYY	AF x ⊠	Deposita in memoria il contenuto di A(x) all'indirizzo diretto C(y)(y); permette l'accesso a 256 byte ovvero 128 parole				FF2E	20	16	FASE α: MEDO=T→BM→NA; 1/8 BC→ MEMORIA; RB=0 — β1
										BF7F	10	00	RA→UA→NA→BA / BB → ECD→MEMORIA — β2
6	AMI	$M_0 \leq (x) \leq 11$ / $A_{12} \leq (x) \leq 15$	A(y)	A8xy	A8xy	Trasferisce in memoria il contenuto del registro A(y) all'indirizzo indiretto contenuto in $M_0 \leq (x) \leq 11$ o in $A_{12} \leq (x) \leq 15$.				3FFF	A0	2C	RA / RB → 1/8 BC → MEMORIA — β1
										FF7F	10	00	RA→UA→NA→BA / BB → ECD→MEMORIA — β2
7	AMIM	$M_0 \leq (x) \leq 11$ / $A_{12} \leq (x) \leq 15$	A(y)	82xy	82xy	Trasferisce in memoria il contenuto del registro A(y) all'indirizzo indiretto contenuto in $M_0 \leq (x) \leq 11$ o in $A_{12} \leq (x) \leq 15$. L'indirizzo contenuto in $M_0 \leq (x) \leq 11$ o in $A_{12} \leq (x) \leq 15$ è decrementato di uno nel registro di partenza.				1F5F	AF	60	RA→UA(-1)→NA→BA→RA; RB→UB→NB→BB→RB; 1/8 BC→ MEMORIA — β1
										FF7F	10	01	RA→UA→NA→BA / BB → ECD→MEMORIA — β2
8	AMIP	$M_0 \leq (x) \leq 11$ / $A_{12} \leq (x) \leq 15$	A(y)	88xy	88xy	Trasferisce in memoria il contenuto del registro A(y), all'indirizzo indiretto contenuto nel registro $M_0 \leq (x) \leq 11$ o in $A_{12} \leq (x) \leq 15$. L'indirizzo contenuto in $M_0 \leq (x) \leq 11$ o in $A_{12} \leq (x) \leq 15$ è incrementato di uno e posto nel registro di partenza.				015F	AF	60	RA→UA(+1)→NA→BA→RA; RB→UB→NB→BB→RB; 1/8 BC→ MEMORIA — β1
										FF7F	10	01	RA→UA→NA→BA / BB → ECD→MEMORIA — β2
9	AND	A(x)	B(y)	97xy	97xy	Posiziona il deviatore D101 in fuznione del risultato dell'AND tra i contenuti dei registri A(x) e B(y); il risultato è trascurato.		=0		BBF2	50	08	RA / RB → UA → D101
10	ANDA	A(x)	B(y)	A7xy	A7xy	Esegue l'operazione AND tra il contenuto dei registri A(x) e B(y); deposita il risultato in A(x). Posiziona il deviatore D101.		=0		BB72	53	18	RA / RB → UA → D101; NA→BA→RA
11	ANDB	A(x)	B(y)	B7xy	B7xy	Esegue l'operazione AND tra il contenuto dei registri A(x) e B(y); deposita il risultato in B(y). Posiziona il deviatore D101		=0		BB92	5C	18	RA / RB → UA → D101; NB→BB→RB
12	AZAM	A(x)		ABxF	ABxF	Azzera i quattro bit meno significativi del registro A(x).				B97F	11	18	UA→NA→BA→RA; UA ← 0
13	AZAP	A(x)		BBxF	BBxF	Azzera i quattro bit più significativi del registro A(x).				B97F	12	18	UA→NA→BA→RA; UA ← 0
14	AZBM	B(x)		CBxF	CBxF	Azzera i quattro bit meno significativi del registro B(x)				797F	14	18	UA→NA→BB→RB; UA ← 0
15	AZBP	B(x)		DBxF	DBxF	Azzera i quattro bit più significativi del registro B(x)				797F	18	18	UA→NA→BB→RB; UA ← 0

LEGENDA

TABELLA DELLE MICROISTRUZIONI

A(x) o A(y) = Come operandi indicano un registro RA di scratch-pad in cui (x) e (y) definiscono i registri interessati nella microistruzione. Si usa (x) per indicare il primo operando e (y) per indicare il secondo.

B(x) o B(y) = Come operandi indicano un registro RB di scratch-pad in cui (x) e (y) definiscono i registri interessati. Si usa (x) per indicare il primo operando e (y) per indicare il secondo.

L(x) o L(y) = Come operandi indicano un registro L di scratch-pad in cui (x) e (y) definiscono i registri interessati. Si usa (x) per indicare il primo operando e (y) per indicare il secondo.

C(x) o C(y) = Costante usata come maschera o come indirizzo diretto di memoria.

M0 ≤ (x) ≤ 11
A12 ≤ (x) ≤ 15 = Indirizzo indiretto di memoria in cui (x) definisce un registro L di scratch-pad che deve essere compreso tra LØØ e L11. Il contenuto di L (16 bit) rappresenta l'indirizzo di memoria. Si usa 0 ≤ (x) ≤ 11 per indicare l'operando 1. Nel caso che si usi un registro superiore a 11, per indirizzare in modo indiretto la memoria, esso viene considerato da hardware come un registro corto e precisamente A con 12 ≤ (x) ≤ 15 mentre l'altra parte di registro, cioè la B, non è presa in considerazione. Su tabulato verrà scritto non M ma A compreso tra 12 e 15. In questo caso con 8 bit si possono indirizzare solo i primi 256 byte di memoria: infatti $2^8 = 256$.

z = Viene usato per indicare un registro indirizzatore di microprogramma cioè: LØØ, LØ1, A12, A13.

⊠ = Significa che il valore binario dei bit non ha importanza per l'esecuzione delle microistruzione: possono essere indifferentemente a 1 oppure a 0.

//8 o //16 = Nel flusso di funzionamento, il registro BC è contraddistinto da //8 se indirizza il byte, da //16 se indirizza la parola.

LIV.4 = 0
LIV.3 = 1
LIV.2 = D
LIV.1 = C

No	NOME	OPERANDO 1	OP. 2	COD. ESAD.	REGIST. R0	DESCRIZIONE	DEVIATORI D1000	DEVIATORI D1010	DEVIATORI D1020	CROM	TROM	VROM B.A	FLUSSO DI FUNZIONAMENTO
1	ADD	A(x)	B(y)	86xy	86xy	Posiziona i deviatori in funzione del risultato della somma [A(x) + B(y) + D100] il cui risultato è trascurato.	>255	=0	>15	93F7	50	08	RA, RB → UA → DEV
2	ADDA	A(x)	B(y)	96xy	96xy	Somma [A(x) + B(y) + D100], deposita il risultato in A(x) e posiziona i deviatori	>255	=0	>15	9377	53	18	D100, RA, RB → UA → DEV; NA → BA → RA
3	ADDB	A(x)	B(y)	A6xy	A6xy	Somma [A(x) + B(y) + D100], deposita il risultato in B(y) e posiziona i deviatori	>255	=0	>15	9397	5C	18	D100, RA, RB → UA → DEV; NB → BB → RB
4	ALFA (∝)	REGISTRO INDIRIZZATORE DI MICROPROGRAMMA: LØØ-LIVELLO 4 LØ1-LIVELLO 3 A13-LIVELLO 2 A12-LIVELLO 1			F0⊠z	La ALFA è una pseudo-microistruzione in quanto il registro R0 non viene caricato da un codice di memoria ma direttamente dall'hardware di unità centrale. Esegue l'indirizzamento della memoria per la lettura di una microistruzione e incrementa di uno il registro indirizzatore di microprogramma.				C15F	3F	60	+1 → RA → UA → NA → BA → RA; RB → UB → NB → BB → RB; BC → MEMORIA → ME00÷14 → R0; ME00÷7 → BM-NA; ME08÷12 → NB (Per P4 SAI)

MARINELLI

olivetti

STAC · Servizio Addestramento Centrale

CPU 19

DESCRIZIONE DI FUNZIONAMENTO

Codice STAC: 801.30.1

Printed in Italy